Orchestrated Intelligence

A blueprint for building enterprise-grade GenAI using Context Engineering

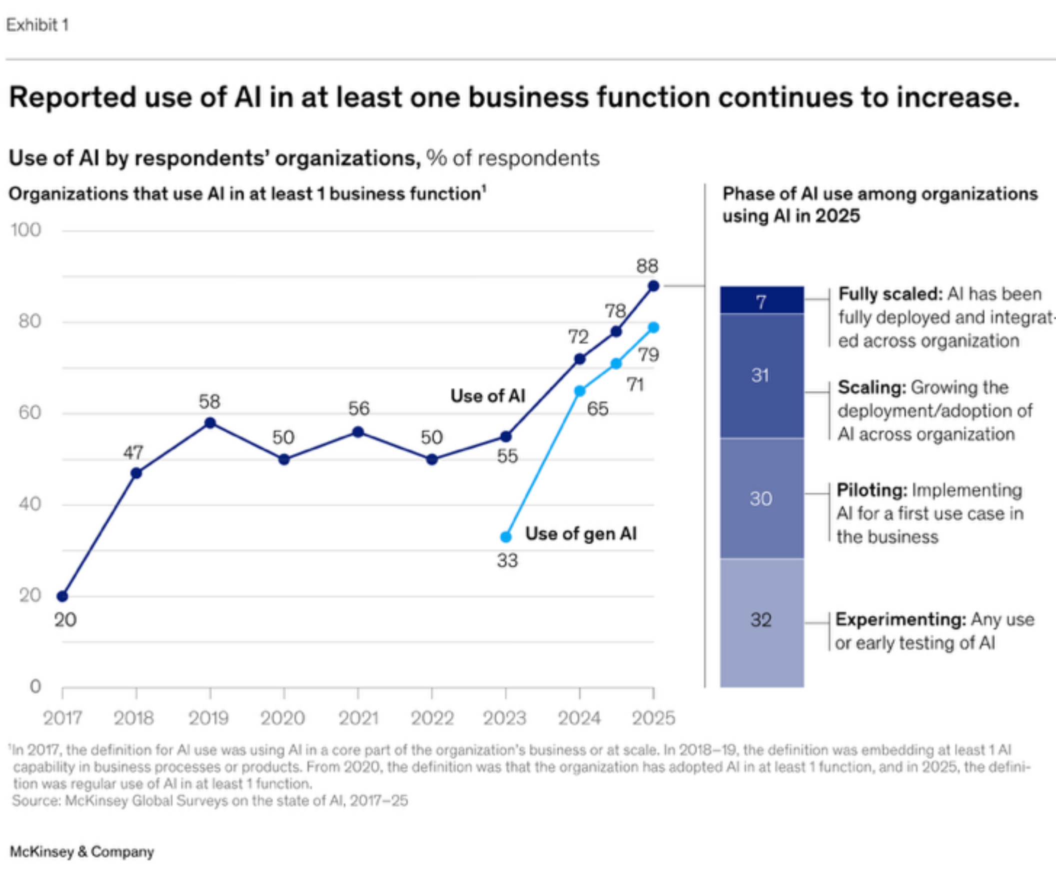
VIANAI

Table of Contents

Introduction

It's a new day, but the story of AI challenges in the enterprise remains surprisingly the same. In a report from November 2025, McKinsey laid out the stakes — nearly two thirds of organizations haven't begun scaling AI across the enterprise. AI remains siloed in specific businesses and teams, and the POCs have not yet given any kind of measurable financial impact.

The McKinsey paper echoes a similar report from MIT, which found that 95 percent of the AI projects in enterprises fail to make it past the POC phase.



While the definitive promise of AI, especially LLMs, may continue these investments in POCs, there continues to be a wide gap between having AI that works in the enterprise, and having AI that is just an interesting display of possibilities.

The age-old adage, “a difference in quantity is a difference in kind” proves true even for enterprises working with AI. The difference between a weekend pet project and a scalable process that works robustly inside a company is immense. The latter requires significant engineering and a complete software system — in addition to the LLMs — while always ensuring accuracy and trust in the responses.

hila is Vianai’s enterprise-grade GenAI and agentic platform, purpose-built to meet the real challenges that have slowed meaningful AI adoption in large organizations. Most GenAI tools fail because they lack domain context, cannot be customized to the customer’s processes, don’t integrate cleanly with existing enterprise systems, and fall apart when held to enterprise standards of security, governance, and robustness. hila is designed from the ground up to solve these exact issues as its agents incorporate deep business logic, adapt to each organization’s workflows, and continuously improve through business user feedback. Rather than imposing a generic AI interface on the user, hila molds itself to the enterprise and its users — respecting existing context, data structures, and access.

As a result, hila is one of the few GenAI platforms that has already delivered meaningful impact. Customers across finance, supply chain and HR rely on hila agents that have been in production for over a year, generating measurable ROI through faster analysis, reduced operational costs, and more confident decision-making. Testimonials highlight capabilities such as conversational finance, advanced simulation and what-if modeling, and automated evaluation of risks and opportunities — all delivered with the accuracy, control, auditability, and security that enterprises require. In a landscape where the vast majority of GenAI pilots never graduate into true value, hila stands out as a platform that brings GenAI into the core of the business: contextual, configurable, integrated, and robust enough for the real world.

> *"The hila Platform is mission critical for the Abudawood Group in today's volatile environment. Its unique, market-leading capabilities in advanced analytics, including real-time simulation and what-if analysis, has super-charged our strategic agility and provided us a real competitive advantage. With just a few strategic questions and simulations, our investment in hila has paid for itself many times over. Using hila, we are dramatically faster, sharper and more informed in our decision-making - proactively responding to market shifts, optimizing resources, and ensuring financial resilience across our entire organization."*

Abdullah AbuSido, Director of Business Applications

hila can answer questions that other GenAI systems cannot, like:

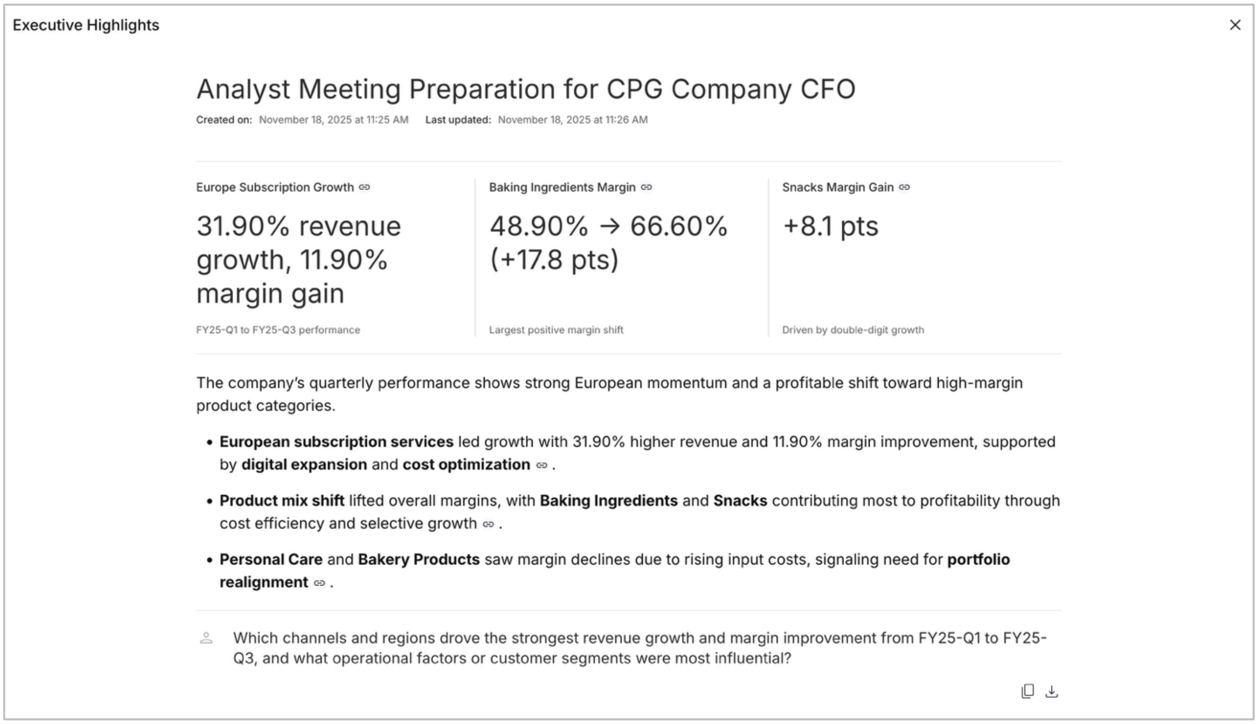
- Based on last year's performance, I need to cut costs. Identify concrete steps to cut costs by 10% without impacting revenue growth.
- Identify irregular patterns in suppliers spend for Q2 2025 based on spend for last quarter compared with spend for last 4 quarters and last 12 quarters.

Many more examples will be outlined throughout this paper.

hila is an end-to-end system, which is essential for delivering autonomous tasks over long time horizons. As the paper Measuring AI Ability to Complete Long Tasks (Kwa et al.) shows, models perform well on tasks that take humans under four minutes, but they falter as complexity and duration increase. In contrast, with robust agentic and systems engineering, hila's Deep Analysis can execute complex tasks that can take humans multiple hours, or even days, in just minutes through deep understanding of the full enterprise context and all the in-built guardrails around accuracy and access control.

In hila, reasoning and context work together in a coordinated interplay: the LLMs contribute their strengths, while the surrounding system augments, constrains, and guides it. As this white paper demonstrates, building a reliable, enterprise-grade agentic system requires designing the entire software stack — not just the LLM — so each component reinforces accuracy, governance, and repeatability.

Below is an example of such a complex analysis question that aids in faster reporting activities – tasks that today can take multiple days to complete:



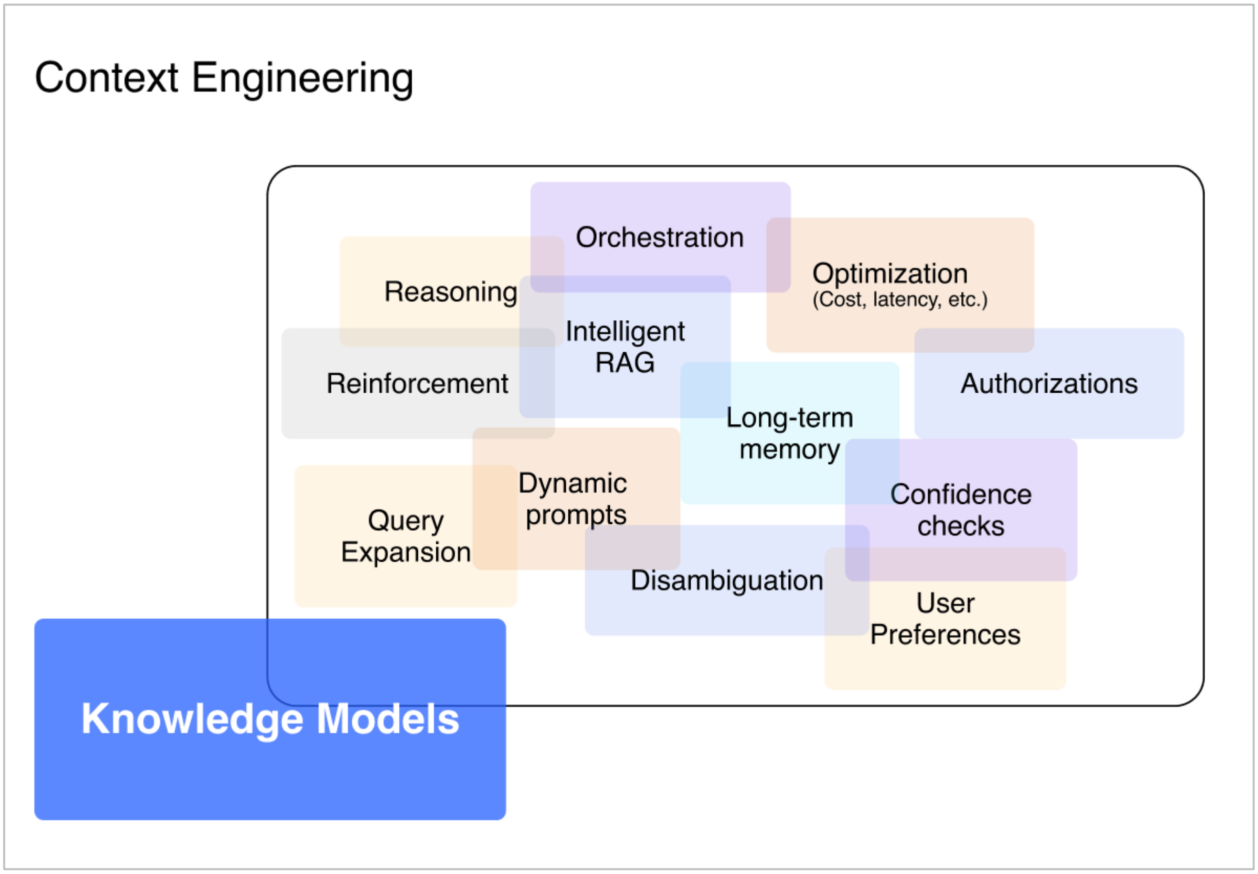
The above is an example of how hila provides analysis in minutes for a process that today can take several days

Ultimately, as we have seen across multiple enterprise deployments at production scale, success only comes from having an *entire* system in place. Only a fully orchestrated architecture turns AI from an impressive demo into a dependable, high-ROI enterprise capability.

GenAI Native Capabilities in hila

GenAI is different, and the capabilities and limitations of LLMs are fundamentally new. For example, GenAI can generate complex code and applications on the fly, enabling fully dynamic and personalized user experiences that were previously impossible. Yet these possibilities also demand strong guardrails to ensure reliability, safety, and alignment with enterprise context. In hila, these guardrails include, among many others, schema-aware retrieval, access-controlled knowledge models, confidence checks, automatic clarification when intent is uncertain, user-specific memory, and reinforcement learning — all of which prevent hallucinations and ensure that outputs remain grounded in enterprise-approved processes and data.

To make these guardrails work in real business environments, a GenAI-native platform must not rely on the LLMSs alone as they will never understand enterprise context that is always based on their proprietary data and processes. What is required is a **context engineering** layer that continuously shapes, constrains, and enriches the model's reasoning. At its core, a GenAI-native solution such as hila relies on the kind of context engineering shown in the chart below — bringing together knowledge modeling, reinforcement learning, disambiguation, orchestration, and authorization rules — as well as guardrail systems that ensure every answer is accurate, governed, and tied to enterprise logic.



All that needs to go into context engineering for accurate answers

Below is an outline of these capabilities, including examples from actual customer case studies where these capabilities have been critical in delivering valuable outcomes, and allowing GenAI to be deployed in production use cases at scale (unlike the challenges and failures faced by POCs mentioned in the MIT report).

Intelligent RAG and Knowledge Modeling

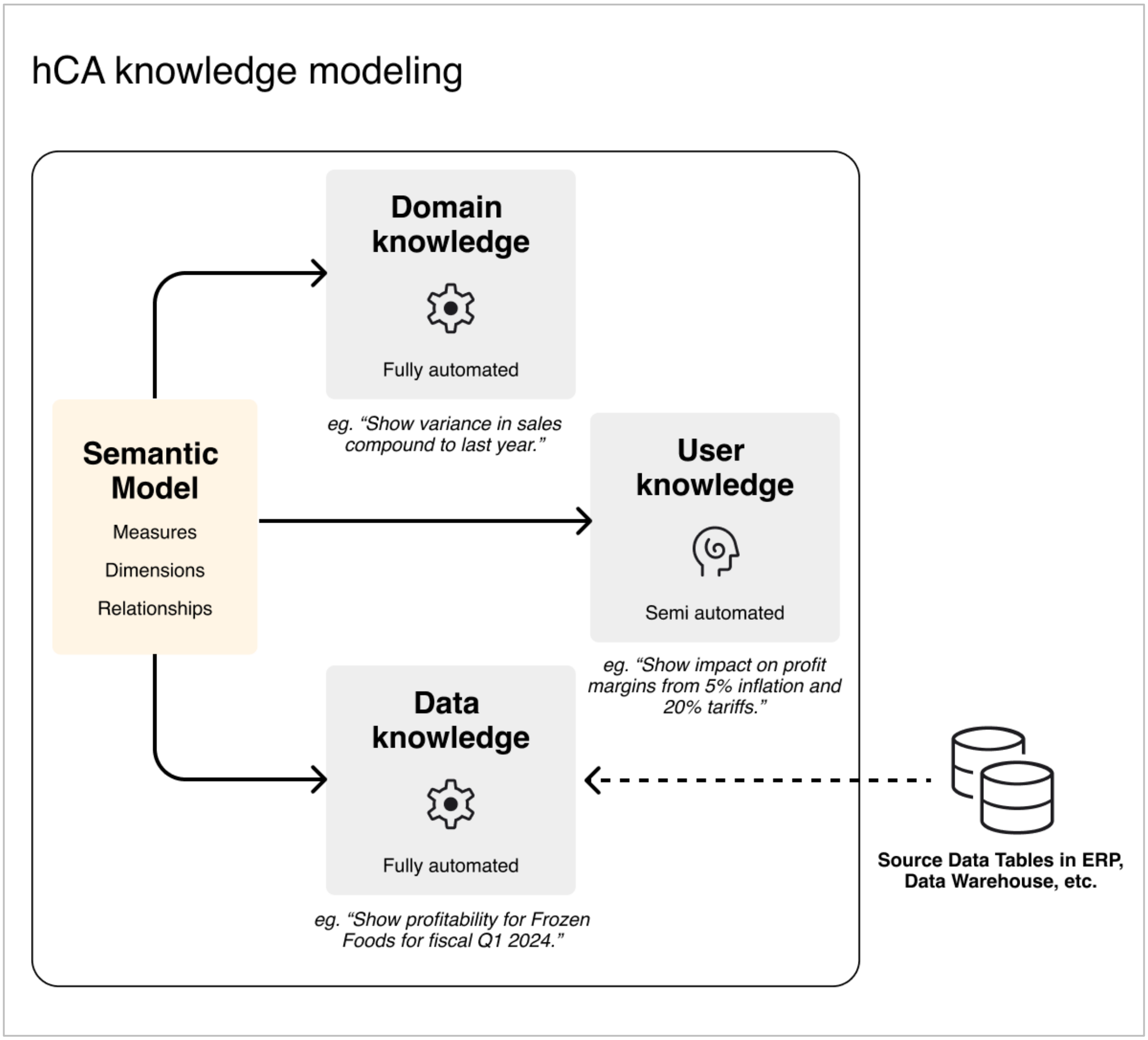
Retrieval-Augmented Generation has been much celebrated as a technique to give LLMs external memory and facts, on top of what facts have already been learnt during training. It's a key part of hila as well, but to additionally support the unique enterprise needs, hila's intelligent RAG must understand data, metadata, logic, and access controls. Without these, RAG becomes just noise rather than intelligent and accurate.

Knowledge modeling

hila builds a knowledge graph that connects all the key artifacts in the enterprise – domain knowledge (e.g. finance metrics), data knowledge (e.g. account hierarchies) and user knowledge (e.g. jargon terms like 'platinum customers'). This ensures semantic alignment — e.g. hila knowing that "gross margin" must connect to specific GL accounts and calculation rules before it can be accurately answered.

	hila	Major Cloud Vendor
Accuracy / completeness	95%+	65%
Responsiveness	13 sec	70 sec+
Deployment time	2 weeks	9 weeks

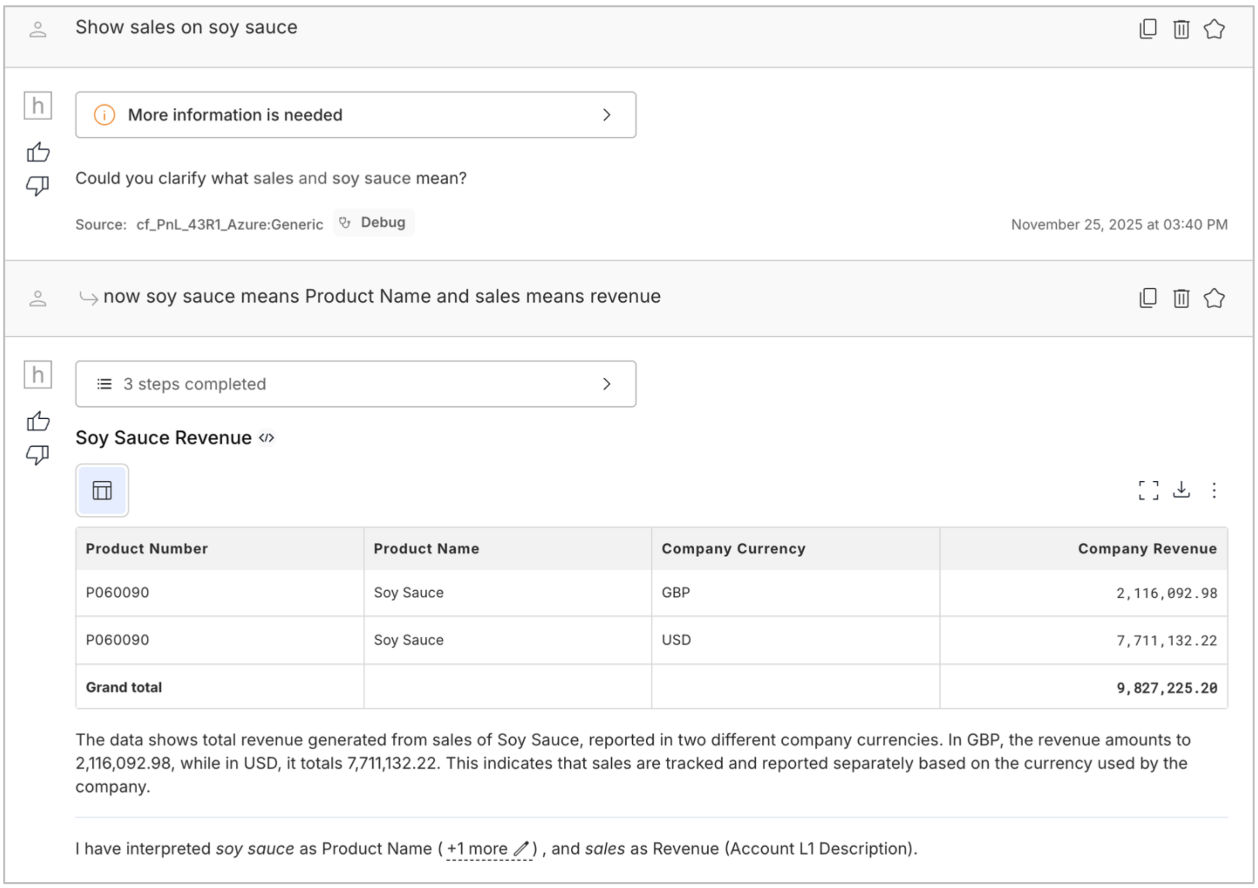
Above is an independent analysis performed by KPMG on behalf of one of our customers. They compared hila with a well-known BI tool from a major cloud vendor. Because of hila's powerful knowledge modeling, it achieved significantly highly accuracy (95% versus 65%), even with far less implementation time (2 weeks versus 9) hila maintains its knowledge in three crucial aspects —company, data and business. When a user submits a query, hila reaches into its knowledge model to ensure the query has the relevant context necessary to generate an accurate response.



Knowledge modeling components of hila, crucial to how it works accurately

Scope validation

Before answering, hila verifies that a query lies within the defined knowledge scope. If not, it declines gracefully rather than hallucinate — preserving trust.



However, hila also provides an ability to grow this knowledge (see Reinforcement Learning later) so that its ability to answer new questions and metrics can be constantly expanded – directly by business users using natural language.

Schema and Domain Awareness

The knowledge model encodes database schema definitions, field meanings, and business logic, enabling domain-grounded instructions. This transforms retrieval into contextual reasoning. In addition, the system has knowledge of the data (including the semantics, measures, dimensions and relationships), and knowledge of the business — such as the context-specific definitions of things. For example, “holiday season” can mean a variety of dates, depending on the country and religious affiliation of the consumer base. This can vary not only from company to company, but inside of a company as well. hila allows this variability in natural language, with customization on the fly.

This is an example of holiday and non-holiday sales:

Holiday vs Non-Holiday Sales

Fiscal Year	Company Currency	Holiday Sales FY23	Non-Holiday Sales FY23	Holiday Sales FY24	Non-Holiday Sales FY24	Holiday Sales FY25	Non-Holiday Sales FY25
FY23	USD	27,656,654.56	235,525,139.91	0.00	263,181,794.47	0.00	263,181,794.47
FY23	GBP	7,940,382.84	68,675,311.68	0.00	76,615,694.52	0.00	76,615,694.52
FY24	USD	0.00	288,222,724.56	30,385,124.16	257,837,600.40	0.00	288,222,724.56
FY24	GBP	0.00	84,534,712.74	9,172,404.05	75,362,308.69	0.00	84,534,712.74
FY25	GBP	0.00	89,616,276.38	0.00	89,616,276.38	8,580,127.59	81,036,148.79
FY25	USD	0.00	310,547,499.35	0.00	310,547,499.35	28,631,634.42	281,915,864.93
Grand total		**35,597,037.40**	**1,077,121,664.62**	**39,557,528.21**	**1,073,161,173.81**	**37,211,762.01**	**1,075,506,940.01**

hila includes reasoning steps, which shows the calculations and retrieval of the information:

Reasoning

Data Retrieved

The query retrieves **fiscal year**, **company currency**, and sales revenue totals for both holiday and non-holiday periods from the **journal**, **calendar**, and **account** tables. Business-relevant columns include fiscal year, company currency, posting date, and company revenue.
It only considers:

- Fiscal years **FY23** through **FY25**
- Accounts where the level 1 description is **Revenue**

Data Calculated

The query performs conditional summations of **company revenue** based on whether the **posting date** falls within or outside the defined **holiday season** for each fiscal year:

- **Holiday Sales FY23**: SUM of company revenue where posting date between 2022-11-25 and 2022-12-31
- **Non-Holiday Sales FY23**: SUM of company revenue where posting date before 2022-11-25 or after 2022-12-31
- **Holiday Sales FY24**: SUM of company revenue where posting date between 2023-11-24 and 2023-12-31
- **Non-Holiday Sales FY24**: SUM of company revenue where posting date before 2023-11-24 or after 2023-12-31
- **Holiday Sales FY25**: SUM of company revenue where posting date between 2024-11-29 and 2024-12-31
- **Non-Holiday Sales FY25**: SUM of company revenue where posting date before 2024-11-29 or after 2024-12-31

The results are grouped by **fiscal year** and **company currency**.
Finally, it sorts the results by:

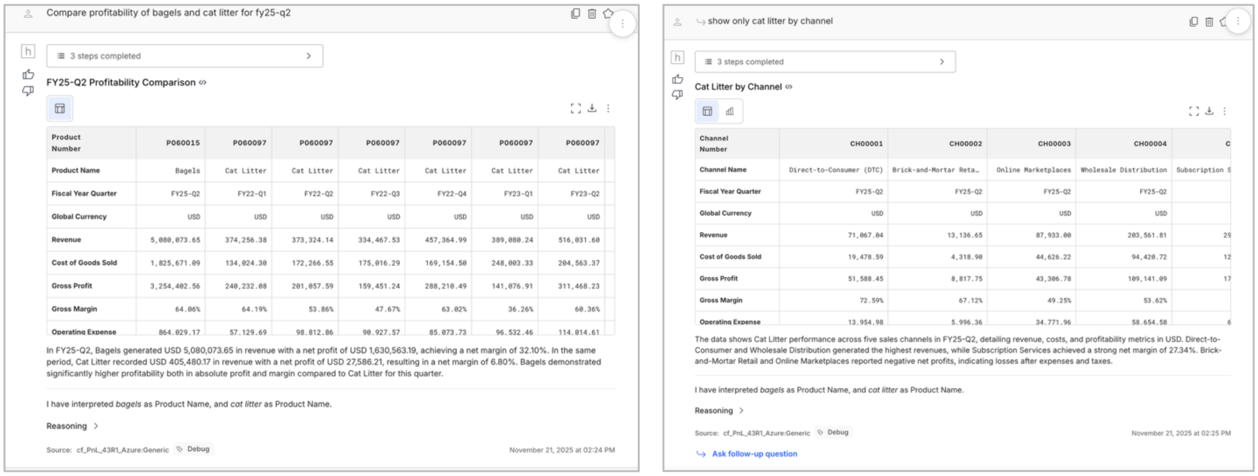
1. **Fiscal Year** — ascending

Working Memory

There are two key components to hila's memory — long and short term — the difference being those that are persistent over many sessions and potentially across many users (long-term), and those that are session specific (short-term). hila also has memory-as-a-tool capabilities, which can decide on things like relevance and importance, based on the query.

Short-term Memory

hila provides session memory through key aspects like disambiguation and follow ups. This enables the user to add additional context to their question, or to redirect the focus of the LLM. This kind of functionality can easily pivot the dimensions returned from a question, and how the analysis is cut.



Example of hila's follow up — on the right, you can see a new cut of the data, with a new dimension, despite that information not being in the original question

Long-term Memory

hila has multiple types of long-term memory. This comes through a combination of defined terms and formulas, both of which are created by a user in natural language. The system then can pull in these definitions on the fly, whenever the user invokes the appropriate term.

These definitions can be complex, requiring multiple calculations, chained calculations and advanced analysis. Similar to macros in an Excel sheet, they are methods to perform calculations, but unlike an Excel sheet, they aren't rigid and can be applied on the fly to a variety of data sources and data types.

The user can verify that a term was disambiguated effectively, and if they want that disambiguation to persist, they can add the term to the lexicon to persist from a short-term memory to long term memory.

Memory as a Tool Retrieval

hila uses an LLM to intelligently infer the intent of the user then invoke the appropriate memory as needed. This is based on a variety of key factors — such as the relevance to the query, the potential accuracy of the output and the necessary information to answer the question correctly.

Disambiguation and Intent Clarification

A key component to answering effectively is in understanding the query, as the user intended it. A system that misunderstands a query can generate accurate data for the wrong question — the worst kind of failure.

Entity Mapping

hila maintains a knowledge model (built automatically and continuously updated based on the data inside enterprise applications) that links natural language to company data concepts (e.g., “plan,” “forecast,” “P&L”). When multiple interpretations exist, hila asks clarifying questions before proceeding.

Entity Memory

Session memory preserves this context for future interactions, while confidence tracking triggers clarifications automatically. These questions understand intent and enable the user to find the correct data on the fly.

One of our customers has a complex organizational hierarchy, with multiple layers from the headquarters to department to sub-department to project. The users do not intuitively use level names like 'headquarters' or 'department.' Rather, they nearly always type the name itself. Moreover, these org names are similar and the same sub-department name exists under different departments. hila disambiguates across all this hierarchical complexity, clarifying the right organization for the user, as they ask their questions about financial performance.

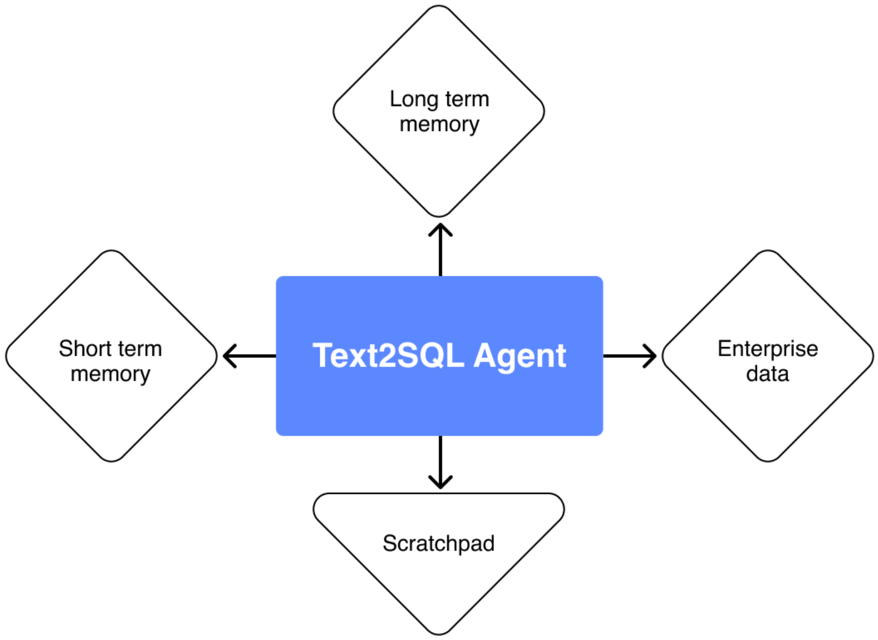
Enhanced Text-to-SQL Framework

Structured data remains the lifeblood of the enterprise — and the hardest to access with natural language. Text-to-SQL (T2SQL) code generation on the fly bridges that gap, but naïve implementations often fail under enterprise schema complexity and business semantics.

hila has enabled one of our customers to create and refine a set of complex finance questions that previously were locked away in SAP Z-Reports. These were slow and inefficient, and insights would have to be pre-created — so no analysis could occur outside of their pre-created reports. This process has been vastly improved through hila's text-to-sql, which directly queries the company's SAP ECC tables with SQL generated on the fly. Some example prompts used to enable this are shown below:

- Variance of amount on local currency for September vs October 2024 by WBS project by vendor, list the ones the variance is over 15k
- Top 5 study codes with highest amount on local currency for GL account 0002005010 in Feb 2024

In addition to many of the capabilities outlined earlier, hila's Text2SQL agent benefits from additional capabilities as below.



Context and Confidence Management

Context is continuously monitored that tracks conversation memory, prunes irrelevant context, and applies confidence scoring to SQL outputs such that low confidence SQL can be rejected to maintain the highest precision.

Post-Processing and Visualization

The generated code is executed, verified, and rendered as structured outputs — charts, summaries, or dashboards. Each result links back to its code lineage, ensuring trust and transparency.

Human-in-the-Loop Reinforcement

There's a last-mile problem that divides an enterprise from the rest of the world. While an LLM can quickly compose a sonnet in the style of a pirate, it cannot understand the information locked behind the walls of an enterprise.

The human experts within an enterprise need the necessary tooling to continuously provide feedback (as the enterprise processes and terms change) to the models. This enables continual learning and 100 percent accurate responses.

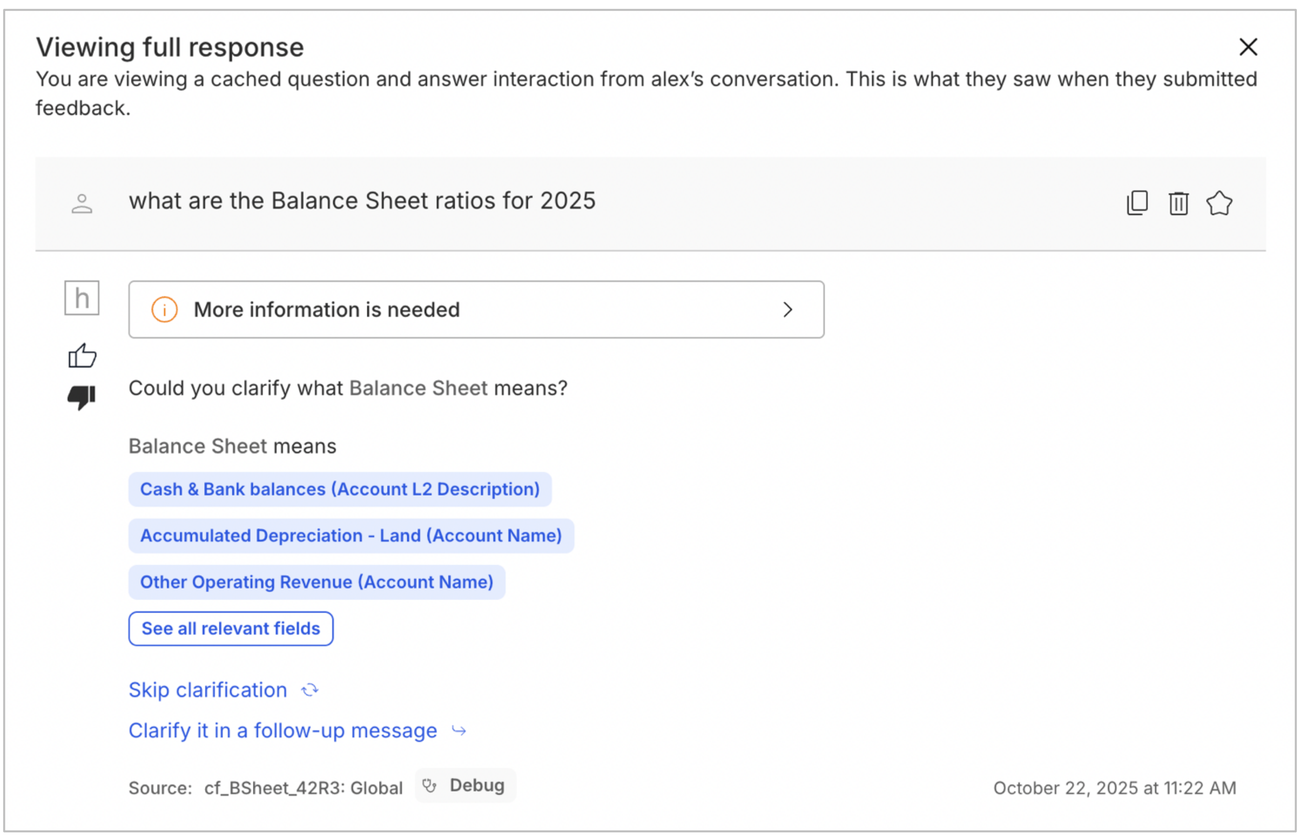
> One of our customers has a "u-report," which comes together on a regular basis. This report is commonly known across every person in the finance org, but outside of the org the "u-report" is a complete mystery. In hila, they've defined "u" simply as "gross margin," and can use hila to process the report on the fly in their language.
>
> In another example, our client has defined a new metric, which is the sum of several other measures and helps track the total work volume at the project, department and division level. This formula is defined as:
>
> **What's the work volume for March 2025?**
>
> **Definition:** work volume = sum(allocation_amount + contract_amount + direct_expense_amount)

Domain Specific Questions

hila allows authorized users to define and verify domain-specific formulas and shorthand through natural language. These definitions augment the knowledge scaffold that aligns LLM reasoning with the enterprise's real vocabulary. For example, "CLV" can mean a specific horizon, which can vary from company to company. hila allows this variability in natural language.

Feedback Loops

Every answer can be rated by experts and reviewed by admins. This feedback is stored, contextualized, and used for reinforcement learning, creating a continuous learning cycle that strengthens over time.



This is an example of the feedback panel that an admin can review

Reasoning Engine

Accessing the data inside of a company requires significant effort, but once accessed, there are incredible possibilities that we can perform with the data. The models have the capability to provide insights and derive narratives from the data. Such a reasoning engine sits inside of hila and enables Deep Analysis, a key feature of our product.

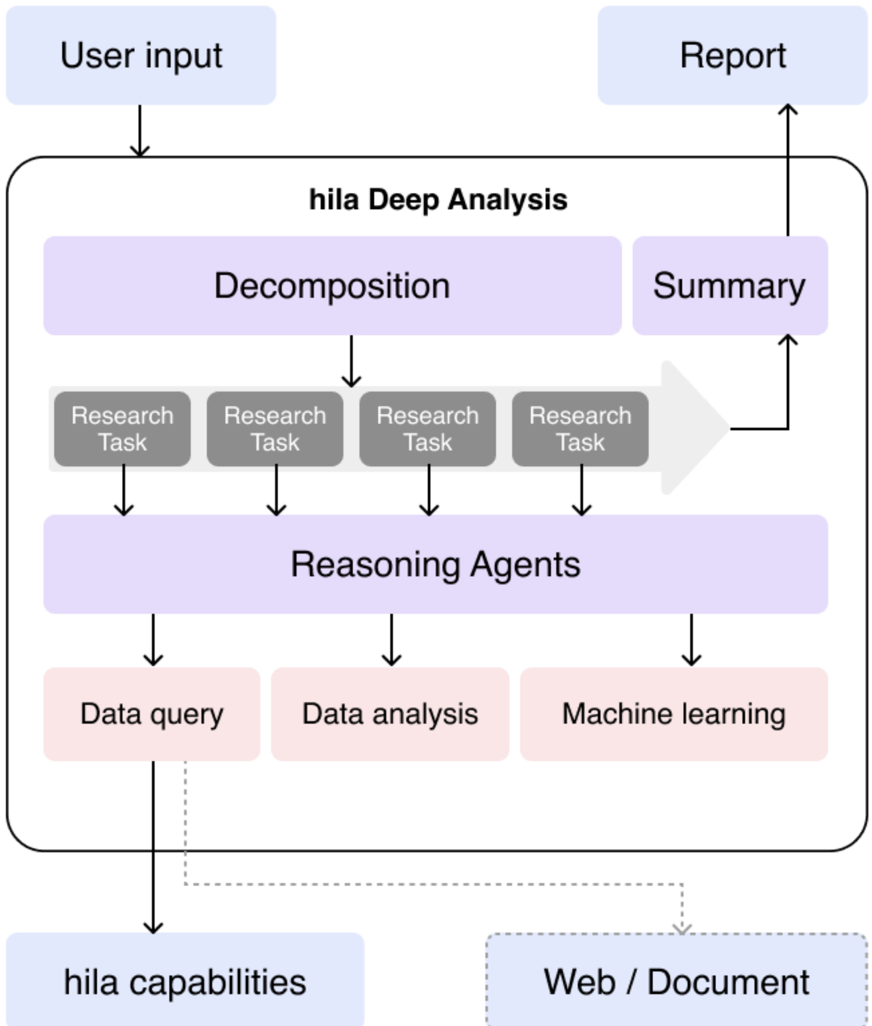
Abudawood has used hila's reasoning engine to understand ramifications around a major supplier pulling out of a key market. With hila, they make strategic decisions in real time, vastly improving their agility and financial insights.

These are example queries that Abudawood used in performing their analysis:

Retrieve gross sales, cost, gross profit and gross profit margin in FY25. Simulate how overall gross profit and gross profit margin change if gross sales and cost of wholesale increases by 10%, gross sales and cost of Value HSM and HSM increases by 15%, gross sales and cost of Pharmacy increases by 20%, the other channels remain the same. Use company currency for all analysis.

Retrieve gross sales, cost and gross profit for FY25 by material level 1. Then build budget scenario with following assumptions: a) for [customer] materials: gross sales keep flat but cost increase by 5% b) for other materials: both gross sales and cost increase by 10%. Using country currency in all calculation. What would be a total gross sales, cost and gross profit before and after?

Retrieve gross sales , cost and gross profit and gross profit margin in Q4 FY25. Then simulate the impact to the company's overall gross profit and profit margin if the gross sales of MPC plant is the same but the cost increased by 10%, keeping the gross sales and cost of non-MPC plant constant. Use country currency for all analysis.



The diagram above shows the architecture for hila's Deep Analysis, including some of the tools that the agentic solution has access to.

Query Understanding

Deep Analysis understands even complex and ambiguous questions and uses the company's knowledge model to break down complex questions into multiple sub-questions and steps to maintain accuracy.

Chain of Reasoning

Generates a reasoning plan and provides continuous updates on the status of the plan to the users, so the user can understand how the final answer is derived. This set of reasoning steps is then executed, which includes code generation, data retrieval, answer review, and summarization.

Interactive & Iterative

Users can interact with the reports to modify breakdown questions and rerun analyses to see the impact on results. Moreover, they can experiment with different questions to see how changes affect results.

Flexible & Scalable

Deep Analysis can work with additional models and agents in various frameworks (MCP, A2A). This enables scaling across an enterprise, utilizing hila's existing layers of knowledge modeling and reinforcement learning, to provide comprehensive answers to complex and ambiguous questions. It can also scale across varying and multiple datasets.

Data Analysis

As a part of the reasoning engine, Deep Analysis has a computation engine, which can interact with the data and generate code to perform advanced calculations, including those specified by the user (such as model or formula type).

Orchestration and Workflow Automation

A single LLM cannot meet the varied demands of enterprise workloads. In our own testing, we found that a cutting-edge foundation model failed to answer enterprise questions against enterprise data 90 percent of the time. To provide value, real-world tasks require multiple models and context engineering working together.

Multi-LLM Coordination

hila's orchestrator routes each request to the LLM best suited for the task, balancing cost, latency, and compliance. For example, a small, fast model might perform Text-to-SQL translation, a domain-tuned LLM may handle summarization, and a private fine-tuned model could generate the executive insights.

Enterprise Integration

hila supports hybrid deployments across local, managed, or customer-specified models (e.g., AWS Bedrock, Azure OpenAI) and different data stores or applications (e.g. SAP, Oracle, Snowflake).

ORACLE	snowflake	databricks	SAP
Global biopharmaceutical leader I Global Natural Gas firm	Multinational publishing firm	International semiconductor and electronics manufacturer I A global materials and coatings company	Middle East consumer goods distributor I European Industrial refrigeration and engineering firm I Japanese technology conglomerate I Global medical organization I European industrial automation and energy multinational I Multinational IT services and consulting enterprise

Above is a sample of hila installations across a variety of conditions and environments, in addition the deployments are on a variety of LLMs, such as Bedrock Anthropic (from Amazon), OpenAI's Azure ChatGPT model, open-source models and Gemini Enterprise.

> *" Vianai is helping customers innovate by bringing its hila agents to Google Cloud. Leveraging the power of Gemini models, these solutions allow businesses to easily deploy sophisticated analytics without technical expertise, unlocking value from their data faster and more effectively."*

Kevin Ichhpurani, President, Global Partner Ecosystem & Channels at Google Cloud

Orchestration and Workflow Automation

LLMs hallucinate. Some consider it a core feature of the models — an aspect that enables creativity. In fact, a recent paper from Thinking Machines considers it a core component of the GPUs that the models are trained and run on.

Examples abound of the variable responses, which are funny in creating a poem, but terrible in providing exact answers that matter. Executives must justify decisions, auditors must verify them, and analysts must understand how results were reached. Explainability turns AI from curiosity into a decision engine.

Transparent Reasoning

hila exposes intermediate logic and data transformations. Each answer includes the underlying query path, applied filters, and calculation steps — so users can verify reasoning rather than assume correctness.

Code and Logic Visibility

Generated code is fully viewable and editable. Users can inspect formulas, measures, and data dimensions, turning opaque LLM reasoning into auditable, reproducible business logic. This transparency creates a feedback loop of trust: users see how conclusions were formed, improving both confidence and accuracy over time.

User Experience and Latency Management

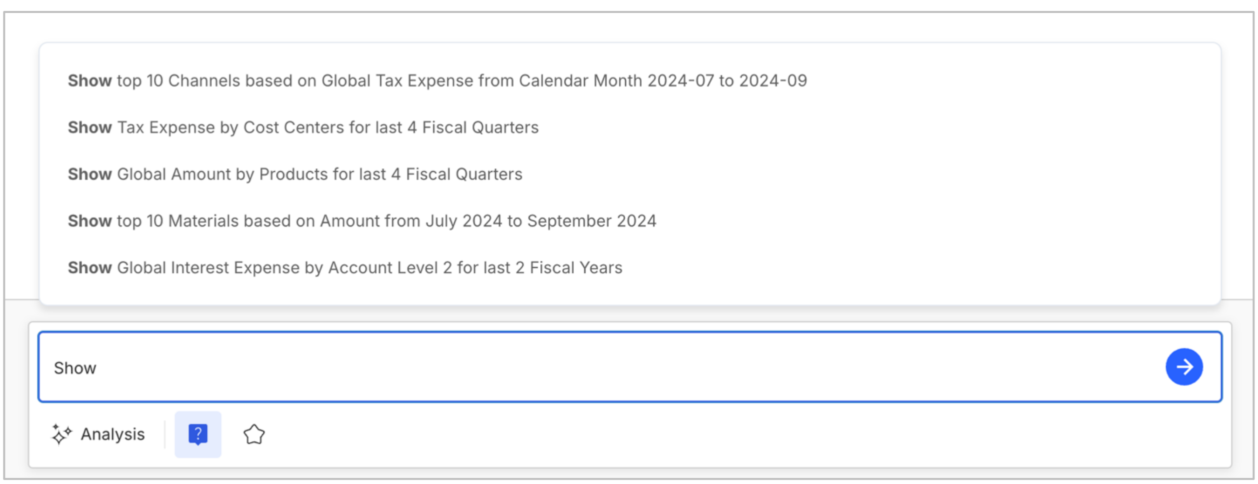
A significant component of adoption is in user experience. Enterprise applications, especially those designed for a business user, should be easy and performant. The consumer sector has put high expectations on simplicity and ease of use, while giving robust results.

Streaming Responses

hila provides a streaming response of tokens, which allows users to see answers unfold as they are computed — reducing latency and maintaining engagement.

Optimistic Interfaces

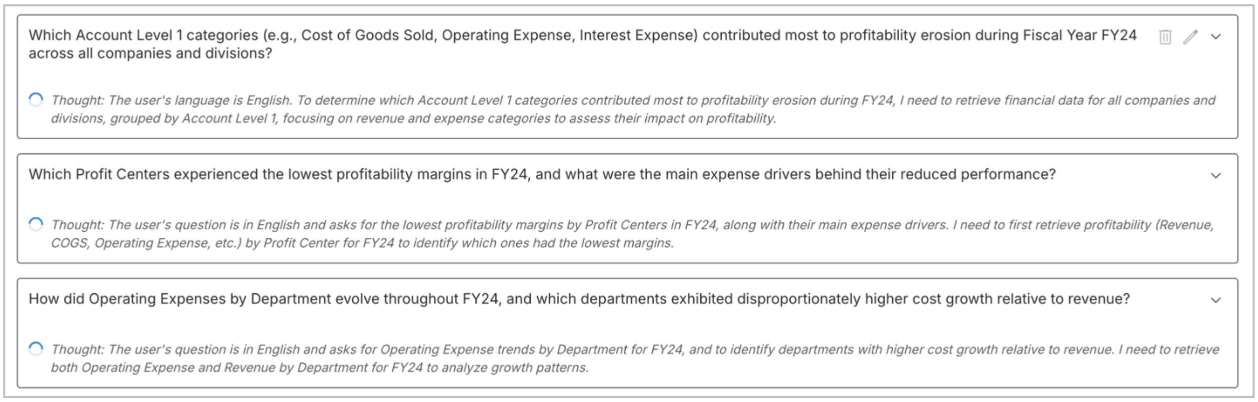
User actions are acknowledged instantly with visual feedback, even before results complete. This responsiveness signals intelligence and control.



Suggested questions are an example of a responsive, intuitive design

Latency-Aware Design

Subtle “thinking” indicators or progress animations make delays feel intentional rather than mechanical. hila’s interface communicates awareness — a small detail that builds disproportionate trust.



The thinking steps are displayed as hila renders the results

Table Stakes for an Enterprise Solution

Aside from the new aspects of a native GenAI solution, there are additional capabilities that are required for it to work in an enterprise. These are non-negotiable, regardless of the enterprise solution. They include security, access control and auditability, for example. hila is enterprise-ready in all these aspects.

Security, Compliance, and Governance

There are key aspects of a platform that are table stakes for an enterprise. These are security, compliance and governance — without which, enterprises would not be willing to adopt an AI platform. For example, no enterprise would be willing to adopt a product that reads their data and then displays the entire compensation to every person in the company. There must be clear guardrails for the data.

> For one of our customers with 2,000+ users, each user can only see the data that they have access to. We control permissions in two ways: the first is through SSO with Microsoft Active Directory to authenticate the login. The second is through controlling datasets by division. Division users can only see the data of their division, and the finance team at headquarters can access data for all divisions.

Role-Based Access and Permissions (RBAC)

hila enforces granular row, column, and object-level policies. Two users can ask the same question and receive different results based on their role, department, region, etc. Preferences cascade through the system — shaping data visibility, UI display, and even model output — ensuring each user sees only what policy allows.

Authentication and Authorization

hila integrates directly with enterprise identity providers via OAuth 2.0 and SSO (Okta, Microsoft, Google, GitHub). Access tokens guarantee that every interaction respects existing security infrastructure — no separate identity layer required.

Data Protection and Encryption

hila encrypts data in transit and at rest using Kubernetes-native mechanisms and cloud encryption APIs. Language model endpoints operate in firewall-protected environments, ensuring no prompts or responses ever traverse unsecured networks.

Auditability and Lineage Tracking

Every query, model input, SQL code, visualization, and summary in hila is logged. Audit trails include the origin of data, assumptions applied, model outputs, and human feedback. Lineage tracking connects input to output — essential for regulated industries. This is available through RBAC to users that have the appropriate access.

Regulatory Compliance

hila's security and operations framework is certified for SOC2 and ISO 27001 standards. This makes it deployable inside finance, manufacturing, and healthcare environments that cannot compromise on compliance.

Scalability and Extensibility

Scaling is one of the most critical problems in an enterprise today. Companies can build proofs of concept, but they can't scale across the enterprise, let alone to different contexts.

Enterprise AI needs to have extensibility to cross functional domains, vertically across workloads, and security across infrastructure — all while controlling cost and performance.

> One of our customers has scaled hila across their entire project management organization — they empower 2,000+ users (soon growing to 5000) with the tool, which maintains all of the fidelity, accuracy and security across this immense scale.

Here is what the CFO of one division initially deploying hila had to say about the future plans:

> *"We see hila as being a transformative tool not just for finance people, but for every P&L owner across the entire company. This would include hundreds of operational managers in every business unit in every division – and we have over 500 business units in our company."*

Model and Infrastructure Flexibility

hila runs equally well in on-prem, hybrid, or cloud environments. It supports concurrent deployment of lightweight models and large general-purpose LLMs, allowing enterprises to scale selectively.

Enterprise Connectors

hila integrates natively with data warehouses, ERPs, and CRMs. Authentication flows, metadata structures, and lineage tracking all extend into existing governance frameworks.

Platform Engineering

Through containerization and Kubernetes orchestration, hila achieves high availability, workload isolation, and automated recovery. This enables a highly flexible deployment architecture.

Intelligent Resource Management

Dynamic routing allocates compute based on task complexity and latency requirements — optimizing both cost and performance in real time.

Error Management and System Verification

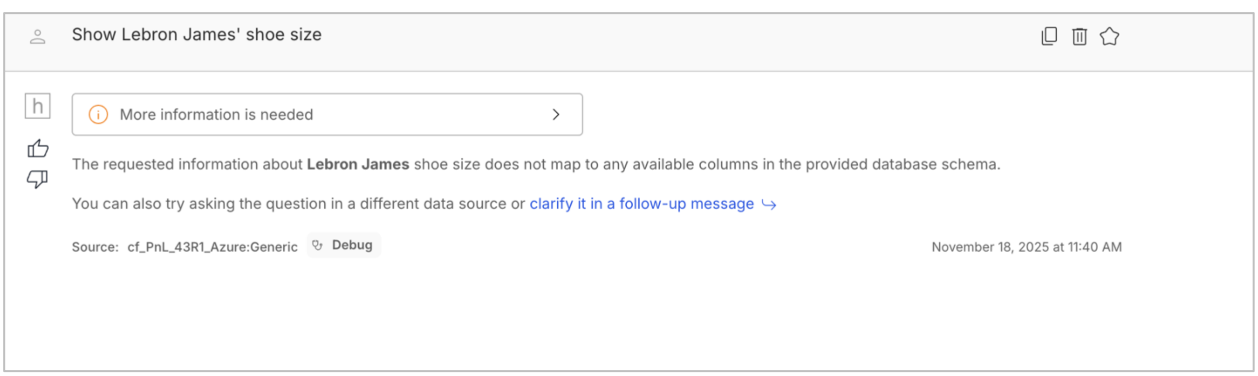
Enterprise systems must not only detect errors but also recover gracefully and learn from them. The system should have the feedback to improve, but also internal mechanisms to provide good, comprehensive answers.

Error Detection and Categorization

hila identifies database connection failures, SQL syntax issues, malformed inputs, or empty responses. Each error type is classified, logged, and tied to its originating process. It alerts the admins proactively and aids in debugging.

Confidence and Fallback Logic

When uncertainty is high, hila doesn't fabricate. It flags the issue, triggers fallback logic (such as rule-based responses or escalation to human review), and clearly communicates uncertainty to the user.



This is an example of a non-response, where hila does not try to make up an answer.

Conclusion

What should be abundantly clear throughout this discussion is that the LLMs are just the start of a full enterprise agentic stack — that the various components necessary for a robust enterprise solution extend far beyond the model itself.

Enterprises no longer need clever chatbots; they need agents that think, reason, and govern. These systems need to be built on top of the foundations in modern enterprise systems — including RBAC, and secure, containerized systems. But also, these systems must have modern advancements available in AI-native platforms, such as text2sql, knowledge modeling and context engineering.

The future belongs to orchestrated intelligence — where models, data, and people operate in tandem under a framework of security, context, and feedback.

Indeed, this is the only way that AI in the enterprise is possible, through careful systems thinking and context engineering. This enables value-added use cases that are secured, explainable, and continuous learning. It then ceases to be a simple POC and becomes the connective tissue of the enterprise itself.